in workspace locations ARG1, ARG2,etc.
The following examples show the method of using commands within a program.

eg1. Programming an A command.

```
                        ;Do a A 1234 0234 command
0D00  21 34 12          LD HL,1234H       ;First argument
0D03  01 34 02          LD BC,0234H       ;Second argument
0D06  DF 41             SCAL  "A          ;Call A command
                        ;
                        ;Do a A 0011 0022 command
0D08  21 11 00          LD HL,0011H       ;First argument
0D0B  01 22 00          LD BC,0022H       ;Second argument
0D0E  DF 41             SCAL "A           ;Call A command
                        ;
                        ;Return to NAS-SYS
0D10  DF 5B             SCAL 5BH
```

eg2. Programming the R,T, and W commands.

This example, which is listed below, begins by writing consecutive hexadecimal numbers into locations 0E00 to 0EFF. This portion of memory is then tabulated using the T command. A prompt is given to start the cassette recorder and the data is written to tape. After waiting for the tape to be rewound, the memory locations are cleared to zero and a second tabulation done to show that the memory is in fact clear. The tape is then read and a third tabulation shows that the tape data has been transferred into memory.

The first call to the T command, line 230, illustrates the standard method of calling a command. The arguments are loaded into workspace locations ARG1 to ARG5. The first three arguments are then transferred to HL,DE, and BC respectively at line 380 using the subroutine ARGS (refer to Chapter 3).

The call to the W command, line 480, is similarly preceded by loading the arguments into ARG1 and ARG2. (There is no copy of these arguments in registers HL and BC since the W command does this transfer within itself.)

The second and third calls to the T command, lines 750 and 940, are abbreviated versions of the first. This is possible because the third, fourth, and fifth arguments remain unchanged in workspace locations ARG3, ARG4, and ARG5; although ARG1 and ARG2 have been changed by the call to the W command this is of no consequence since the T routine takes the first two arguments from HL and DE which are loaded immediately before the call.

The call to the R command, line 850, is a little different to the others. This is because both the R and V commands call up the same routine, the only functional difference being whether or not the input data is transferred to ram. The common routine checks

to see if the command was R or V to determine whether or not to make the transfer. The ASCII code for the command must therefore be placed in workspace location ARGX and, if no argument is used with the R command, zero must be placed in workspace location ARGN.

Listing of example 2.

```
2D00                0010         ORG  2D00H
                    0020 ;NAS-SYS ROUTINE NUMBERS:
2D00 0060           0030 ARGS    EQU  60H
2D00 0061           0040 KBD     EQU  61H
2D00 005B           0050 MRET    EQU  5BH
2D00 0028           0060 PRS     EQU  28H
                    0070 ;NAS-SYS WORKSPACE:
2D00 0C0C           0080 ARG1    EQU  0C0CH
2D00 0C0E           0090 ARG2    EQU  0C0EH
2D00 0C10           0100 ARG3    EQU  0C10H
2D00 0C12           0110 ARG4    EQU  0C12H
2D00 0C0B           0120 ARGN    EQU  0C0BH
2D00 0C2B           0130 ARGX    EQU  0C2BH
2D00 0C14           0140 ARG5    EQU  0C14H
                    0150 ;FILL LOCATIONS 0E00 TO 0EFF WITH
                    0160 ; BINARY COUNT.
2D00 0600           0170         LD   B,0
2D02 21000E         0180         LD   HL,0E00H
2D05 75             0190 FILL    LD   (HL),L
2D06 23             0200         INC  HL
2D07 10FC           0210         DJNZ FILL
                    0220 ;
                    0230 ;DO A T 0E00 0F00 0000 00 0000 COMMAND.
                    0240 ;PUT FIRST ARGUMENT INTO ARG1.
2D09 21000E         0250         LD   HL,0E00H
2D0C 220C0C         0260         LD   (ARG1),HL
                    0270 ;PUT SECOND ARGUMENT INTO ARG2.
2D0F 21000F         0280         LD   HL,0F00H
2D12 220E0C         0290         LD   (ARG2),HL
                    0300 ;PUT THIRD ARGUMENT INTO ARG3.
2D15 210000         0310         LD   HL,0       ;ARG3,ARG4,ARG5
2D18 22100C         0320         LD   (ARG3),HL
                    0330 ;PUT FOURTH ARGUMENT INTO ARG4.
2D1B 22120C         0340         LD   (ARG4),HL
                    0350 ;PUT FIFTH ARGUMENT INTO ARG5.
2D1E 22140C         0360         LD   (ARG5),HL
                    0370 ;MOVE ARGUMENTS INTO REGISTERS.
2D21 DF60           0380         SCAL ARGS
                    0390 ;NOW CALL THE TABULATE COMMAND.
2D23 DF54           0400         SCAL "T
                    0410 ;
                    0420 ;TELL USER TO START RECORDER.
2D25 EF             0430         RST  PRS
2D26 53746172       0440         DEFM /Start recorder now./
     74207265
     636F7264
     6572206E
     6F772E
```

Listing of example 2, continued.

```
2D39 0D00      0450          DEFB 0DH,0
               0460 ;
               0470 ;
               0480 ;DO A W 0E00 0F00 COMMAND.
               0490 ;PUT FIRST ARGUMENT INTO ARG1.
2D3B 21000E    0500          LD   HL,0E00H
2D3E 220C0C    0510          LD   (ARG1),HL
               0520 ;PUT SECOND ARGUMENT INTO ARG2.
2D41 21000F    0530          LD   HL,0F00H
2D44 220E0C    0540          LD   (ARG2),HL
               0550 ; NOW CALL THE W COMMAND.
2D47 DF57      0560          SCAL "W
               0570 ;
2D49 EF        0580          RST  PRS
2D4A 57726974  0590          DEFM /Write done./
     6520646F
     6E652E
2D55 0D        0600          DEFB 0DH        ;NEW LINE ON DISPLAY
2D56 52657769  0610          DEFM /Rewind tape then press <space>./
     6E642074
     61706520
     7468656E
     20707265
     7373203C
     73706163
     653E2E
2D75 00        0620          DEFB 0
               0630 ;WAIT FOR AN INPUT.
2D76 DF61      0640 WAIT     SCAL KBD     ;READ KEYBOARD
2D78 30FC      0650          JR   NC,WAIT
               0660 ;
               0670 ;CLEAR 0E00 TO 0EFF.
2D7A AF        0680          XOR  A
2D7B 47        0690          LD   B,A
2D7C 21000E    0700          LD   HL,0E00H
2D7F 77        0710 CLEAR    LD   (HL),A
2D80 23        0720          INC  HL
2D81 10FC      0730          DJNZ CLEAR
               0740 ;
               0750 ;DO A T 0E00 0F00 0000 00 0000 COMMAND.
2D83 21000E    0760          LD   HL,0E00H     ;FIRST ARGUMENT
2D86 11000F    0770          LD   DE,0F00H     ;SECOND ARGUMENT
               0780 ;Other arguments have not changed.
2D89 DF54      0790          SCAL "T
               0800 ;
2D8B EF        0810          RST  PRS
2D8C 53746172  0820          DEFM /Start tape now./
     74207461
     7065206E
     6F772E
2D9B 0D00      0830          DEFB 0DH,0
               0840 ;
```

Listing of example 2, continued

```
                0850 ;DO AN R COMMAND
                0860 ;PUT ASCII CODE FOR R INTO ARGX.
2D9D 3E52       0870        LD   A,52H
2D9F 322B0C     0880        LD   (ARGX),A
                0890 ;PUT ZERO INTO ARGN; NO ARGUMENTS FOR R.
2DA2 AF         0900        XOR  A
2DA3 320B0C     0910        LD   (ARGN),A
                0920 ;CALL THE R COMMAND
2DA6 DF52       0930        SCAL "R
                0940 ;DO A T 0E00 0F00 0000 00 0000 COMMAND.
2DA8 21000E     0950        LD   HL,0E00H
2DAB 11000F     0960        LD   DE,0F00H
2DAE DF54       0970        SCAL "T
                0980 ;
                0990 ;RETURN TO NAS-SYS.
2DB0 DF5B       1000        SCAL MRET
```

2.4 Changing the Commands.

There is no difference in the way commands and subroutines are processed by NAS-SYS. Both are accessed from a subroutine table which lists the start addresses of both commands and routines.

2.4.1 How NAS-SYS accesses the routines

Each command and routine has a number which indicates its position in the subroutine address table which begins at location STABA (0782,NS3; 0788,NS1). The table starts with the 'A' command (routine number 41H) and the first two locations of the table give the address at which the 'A' command routine begins. Since the first routine is numbered 41H (this being convenient since it is the ASCII code for A), the conceptual start of the table (ie. the table base address, STAB) is 82H less than STABA, ie. 0782 - 82 = 0700 in NS3 or 0788-82=0706 in NS1.
The program instruction SCAL "A (= SCAL 41H = RST 18H:DEFB 41H = DF 41) causes NAS-SYS to read the byte 41H, multiply it by two and add it to the table base address. The result is the location of the 'A' command in the table, which gives the address of the start of routine number 41H. NAS-SYS then jumps to that address so executing the required routine.

As another example, suppose SCAL MRET (=SCAL 5BH) is programmed. NAS-SYS 3 calculates STAB + 2 x 5B,
ie. 0700 + 2 x 5B = 07B6.
The two bytes at location 07B6/7 are the address of the start of the MRET routine; NAS-SYS then jumps to this address and so executes MRET.

The address of the start of the table of routines, ie. STAB is held in workspace location $STAB (0C71/2). The memory space is thus:

Location	Contents
$STAB (0C71)	Address of the conceptual start of table,
(0C72)	normally STAB = STABA-82 = 0700(NS3),0706(NS1)
	points to
STAB normally	Conceptual start of table.
	.
	.
	.
STABA normally	Address of A command, routine number 41H.
	Address of B command, routine number 42H.
	.
	.
	.
	Address of CPOS, routine number 7CH.